

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2001-101396

(43)Date of publication of application : 13.04.2001

(51)Int.Cl.

G06T 3/00
H04N 5/225
H04N 5/228
H04N 5/907

(21)Application number : 11-280500

(71)Applicant : TOSHIBA CORP

(22)Date of filing : 30.09.1999

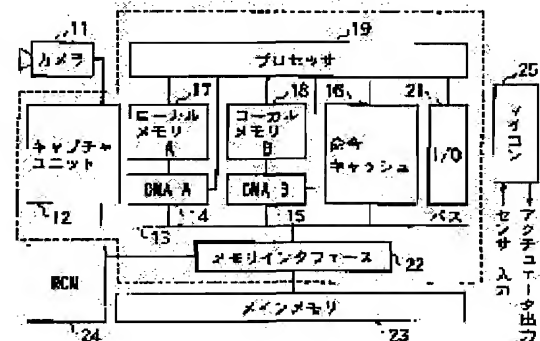
(72)Inventor : ASANO SHIGEHIRO

(54) PROCESSOR AND METHOD FOR CORRECTING IMAGE DISTORTION AND MEDIUM WITH PROGRAM PERFORMING IMAGE DISTORTION CORRECTION PROCESSING STORED THEREIN

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a device which corrects the distortion of an image fetched by a video camera, especially image distortion caused by an image pickup lens.

SOLUTION: This processor comprises a capture unit 12 fetching an image from a camera 11, local memories 17 and 18 to which the image fetched by the capture unit 12 is transferred, a processor 19 which accesses the local memories and performs image distortion correction processing of the image transferred to the local memories, and DMAs 14 and 15 which perform DMA transfer of the image onto the local memories according to a DMA transfer instruction string which is preliminarily prepared to convert the address of the image that is not subjected to distortion correction processing yet.



(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開2001-101396

(P2001-101396A)

(43)公開日 平成13年4月13日(2001.4.13)

(51)Int.Cl.⁷

識別記号

F I

テーマコード(参考)

G 0 6 T 3/00

H 0 4 N 5/225

C 5 B 0 5 7

H 0 4 N 5/225

5/228

Z 5 C 0 2 2

5/228

5/907

B 5 C 0 5 2

5/907

G 0 6 F 15/66

3 6 0

審査請求 未請求 請求項の数 8 O L (全 9 頁)

(21)出願番号

特願平11-280500

(71)出願人 000003078

株式会社東芝

神奈川県川崎市幸区堀川町72番地

(22)出願日

平成11年9月30日(1999.9.30)

(72)発明者 浅野 滋博

神奈川県川崎市幸区小向東芝町1番地 株

式会社東芝研究開発センター内

(74)代理人 100058479

弁理士 鈴江 武彦 (外6名)

Fターム(参考) 5B057 CA08 CA12 CA16 CB08 CB12

CB16 CD12 CE05 CH11 CH14

CH16

5C022 AA01 AB51

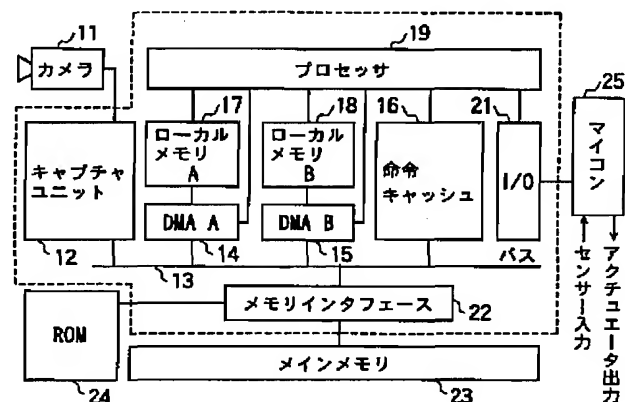
5C052 AA17 GA01 GB01 GE04 GF04

(54)【発明の名称】 画像歪み補正処理装置および方法、並びに画像歪み補正処理を行うプログラムを格納した媒体

(57)【要約】

【課題】本発明は、ビデオカメラにより取り込んだ画像の歪み、特に撮像レンズに起因する画像の歪みを補正する装置を提供する。

【解決手段】カメラ11からの画像を取り込むキャプチャユニット12と、この画キャプチャユニットで取り込まれた画像が転送されるローカルメモリ17,18と、このローカルメモリへアクセスしてこのローカルメモリに転送された画像の歪み補正処理を行うプロセッサ19と、歪み補正処理前の画像のアドレスを変換するために予め作成されたDMA転送命令の列にしたがって画像をローカルメモリ上にDMA転送するDMA14,15とで構成される。



【特許請求の範囲】

【請求項 1】 撮像手段より取り込んだ画像の歪みを補正する画像歪み補正処理装置において、前記撮像手段から画像を取り込む画像取り込み部と、この画像取り込み部で取り込まれた前記画像が転送される内部メモリと、この内部メモリへアクセスして該内部メモリに転送された前記画像の歪み補正処理を行うプロセッサと、前記歪み補正処理前の前記画像のアドレスを変換するために予め作成された DMA 転送命令の列にしたがって該画像を前記内部メモリ上に DMA 転送する DMA 部と、を有することを特徴とする画像歪み補正処理装置。

【請求項 2】 撮像手段により撮像された画像を外部メモリに取り込む画像取り込みユニットと、前記外部メモリへのインターフェースユニットと、前記外部メモリに取り込まれた前記画像が所定の単位で転送される内部メモリと、この内部メモリへアクセスして該内部メモリに転送された前記画像の歪み補正処理を行うプロセッサと、前記歪み補正処理前の前記画像のアドレスを変換するために予め作成された DMA 転送命令の列にしたがって該画像を前記内部メモリ上に DMA 転送する DMA ユニットと、前記歪み補正処理後の前記画像を外部へ出力する I/O ユニットとを 1 チップで構成したことを特徴とする画像歪み補正処理装置。

【請求項 3】 前記 DMA 部は、前記歪み補正処理後の前記画像のアドレスの変化がリニアになるように該歪み補正処理前のアドレスを変換するように予め作成された DMA 転送命令の列に従って該画像を前記内部メモリに転送することを特徴とする請求項 1 または 2 に記載の画像歪み補正処理装置。

【請求項 4】 前記 DMA 部は、前記内部メモリに格納された DMA 転送のシーケンスを記述したディスクリプトにより起動されることを特徴とする請求項 1 ないし 3 のいずれか 1 に記載の画像歪み補正処理装置。

【請求項 5】 前記内部メモリは少なくとも 2 つのメモリを有する複数バッファ構造を有し、前記プロセッサは一のメモリに画像を転送中に他のメモリの内容を処理する請求項 1 ないし 4 のいずれか 1 に記載の画像歪み補正処理装置。

【請求項 6】 前記プロセッサは、歪みを伴った前記画像から 1 ドットづつ該画像を読み込み、各ドットの歪み補正処理を行い、歪み補正処理後のドットを周辺のドットとの重みつき平均値を演算し、平均化処理したドットを前記内部メモリに書き込むことを特徴とする請求項 1 ないし 5 のいずれか 1 に記載の画像歪み補正処理装置。

【請求項 7】 撮像手段より取り込んだ画像の歪みを補正する画像歪み補正処理方法において、前記撮像手段から画像を取り込むステップと、

この画像取りステップで取り込まれた前記画像を内部メモリに転送するステップと、前記内部メモリへアクセスして該内部メモリに転送された前記画像の歪み補正処理を行うステップと、前記歪み補正処理前の前記画像のアドレスを変換するために予め作成された DMA 転送命令の列にしたがって該画像を前記内部メモリ上に DMA 転送するステップと、を有することを特徴とする画像歪み補正処理方法。

【請求項 8】 撮像手段から画像を取り込む命令と、この命令によって取り込まれた前記画像を内部メモリに転送する命令と、前記内部メモリへアクセスして該内部メモリに転送された前記画像の歪み補正処理を行う命令と、前記歪み補正処理前の前記画像のアドレスを変換するために予め作成された DMA 転送命令の列にしたがって該画像を前記内部メモリ上に DMA 転送する命令と、を有するプログラムを格納した媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、歪みを含むカメラで取り込んだ画像を、処理に適するような歪みの補正された画像に修正する技術に関する。

【0002】

【従来の技術】 近年、画像を計算機により認識し、結果を提示する装置は、監視や、セキュリティの分野で重要度を増している。一般にこのような画像認識装置はカメラにより画像を取り込み、これを計算機に入力するときに直交する座標軸を用いた配列の形のデータにして取り込む。例えば、パーソナルコンピュータでは標準的な画像として VGA 画像が使用され、これは縦 400 ドット、横 640 ドットのマトリックスとして構成されている。

【0003】 一方、カメラから取り込んだ画像はレンズの歪みなどにより、理想的な画像から比べると歪みを持っており、広角レンズを使用した場合には歪みが顕著に現れる。特に、広角レンズを備えたカメラで画像を取り込んだ場合、カメラから遠い画像がカメラに近づくに従って画像に歪みが生じ、遠距離の画像と近距離の画像とでは歪みにより異なる画像となってしまう。

【0004】 上記のようなカメラにより得られた画像を計算機により画像認識を行うアルゴリズムでは、動きベクトルの検出を行うような処理が広く行われている。この動きベクトルは、現在の画像を、過去の画像と比較し、過去の画像の中のあるオブジェクトが現在の画像の中のあるオブジェクトに対して縦横に移動している事を検知することによって検出する。また、一般に動きを検出する場合には画像の空間方向の微分成分（エッジ）を検出することにより処理に必要な情報を減らして処理する事も多い。動いているオブジェクトのエッジを過去の画像のエッジと比較し、その動きベクトルから動きを認

識する。

【0005】画像の動きベクトルを検出する場合、画像に歪みがあると実際にはオブジェクトは平行に移動しているにも関わらず歪みを受けて移動したように処理されたり、あるいはエッジの動きを検出しようとしたときにエッジが移動するに伴ってエッジの角度（縦横の比）が変化して動きの検出が複雑になるなどの問題点が生じる。

【0006】

【発明が解決しようとする課題】動画画像を認識する場合、オブジェクトのエッジ、例えば明るさが大きく変化している部分を抽出し、それを線として捕らえる。オブジェクトの例えば垂直エッジを過去画像と現画像とで比較することによってオブジェクトが動いているかどうか

が認識できる。

【0007】しかし、カメラレンズに起因して画像が歪む場合、例えば広角レンズで撮像された場合、オブジェクトはカメラに近づくにつれてそのエッジがレンズの歪みにより傾く。このため、オブジェクトは過去画像と現画像とは同じものだと認識されなくなる。そこで、広角

レンズによる画像の歪みを補正する必要がある。

【0008】この画像の歪みを補正し、処理する装置としては一般にマイクロプロセッサが使用される。マイクロプロセッサでは処理を高速化するためにキャッシュメモリを使用しているが、画像のサイズは一般にキャッシュメモリの容量よりも大きいので、キャッシュが有効に働かず、高速化を阻んでいた。

【0009】本発明は、上記問題点に鑑み、画像の歪み、特に撮像レンズに起因する画像の歪みの補正をマイクロプロセッサで効率的に行う画像歪み補正処理装置および方法並びに画像歪み補正処理を行うプログラムを格納した媒体を提供する。

【0010】

【課題を解決するための手段】この発明は、撮像装置より取り込んだ画像の歪みを補正する画像歪み補正処理装置において、撮像装置から画像を取り込む画像取り込み部と、この画像取り込み部で取り込まれた画像が転送される内部メモリと、この内部メモリへアクセスして内部メモリに転送された画像の歪み補正処理を行うプロセッサと、歪み補正処理前の画像のアドレスを変換するために予め作成されたDMA転送命令の列にしたがって画像を内部メモリ上にDMA転送するDMA部とを有することを特徴とする画像歪み補正処理装置を提供する。

【0011】本発明によると、撮像画像に歪みを生じさせる広角レンズなどを備えたビデオカメラのような撮像装置により取り込まれる画像に歪み補正処理を施すことにより画像を汎用プロセッサにより処理しやすくなる。また、画像はローカルメモリにDMA転送を行い、ローカルメモリの画像に対して歪み補正を行うので高速画像処理が可能となる。

【0012】また、この発明は、撮像装置により撮像された画像を外部メモリに取り込む画像取り込みユニットと、外部メモリへのインターフェースユニットと、外部メモリに取り込まれた画像が所定の単位で転送される内部メモリと、この内部メモリへアクセスして内部メモリに転送された画像の歪み補正処理を行うプロセッサと、歪み補正処理前の画像のアドレスを変換するために予め作成されたDMA転送命令の列にしたがって画像を内部メモリ上にDMA転送するDMAユニットと、歪み補正処理後の画像を外部へ出力するI/Oユニットとを1チップで構成したことを特徴とする画像歪み補正処理装置を提供する。

【0013】本発明によると、画像取り込み部と画像処理部が1チップで構成されるので処理の高速化並びに装置の小型化が実現できる。また、外部メモリに対して直接アクセスをするのではなく、外部メモリから内部メモリにDMA転送し、内部メモリに格納された画像に対してプロセッサにより画像歪み補正処理を行うので、簡単に高速なアクセスが可能となり、これに伴って画像歪み補正処理の高速化が実現できる。

【0014】また、この発明は、上記のDMA部が、歪み補正処理後の画像のアドレスの変化がリニアになるように歪み補正処理前のアドレスを変換するように予め作成されたDMA転送命令の列に従って画像を内部メモリに転送することを特徴とする画像歪み補正処理装置を提供する。

【0015】また、この発明は、上記DMA部が、内部メモリに格納されたDMA転送のシーケンスを記述したディスクリプタにより起動されることを特徴とする画像歪み補正処理装置を提供する。

【0016】また、この発明は、上記の内部メモリが少なくとも2つのメモリを有する複数バッファ構造を有し、上記のプロセッサが一のメモリに画像を転送中に他のメモリの内容を処理する画像歪み補正処理装置を提供する。

【0017】また、この発明は、上記のプロセッサが、歪みを伴った前記画像から1ドットづつ該画像を読み込み、各ドットの歪み補正処理を行い、歪み補正処理後のドットを周辺のドットとの重みつき平均値を演算し、平均化処理したドットを前記内部メモリに書き込むことを特徴とする画像歪み補正処理装置を提供する。

【0018】また、この発明は、撮像装置より取り込んだ画像の歪みを補正する画像歪み補正処理方法において、撮像装置から画像を取り込むステップと、この画像取りステップで取り込まれた前記画像を内部メモリに転送するステップと、内部メモリへアクセスして内部メモリに転送された前記画像の歪み補正処理を行うステップと、歪み補正処理前の画像のアドレスを変換するために予め作成されたDMA転送命令の列にしたがって画像を内部メモリ上にDMA転送するステップとを有すること

を特徴とする画像歪み補正処理方法を提供する。

【0019】また、この発明は、撮像装置から画像を取り込む命令と、この命令によって取り込まれた前記画像を内部メモリに転送する命令と、内部メモリへアクセスして該内部メモリに転送された前記画像の歪み補正処理を行う命令と、歪み補正処理前の前記画像のアドレスを変換するために予め作成されたDMA転送命令の列にしたがって該画像を前記内部メモリ上にDMA転送する命令とを有するプログラムを格納した媒体を提供する。

【0020】

【発明の実施の形態】図1は、この発明の一実施形態に従った画像歪み補正処理装置を示している。これによると、オブジェクトを撮像する、例えば広角レンズを備えたビデオカメラ（撮像手段）11がキャプチャユニット12を介してバス13に接続される。このバス13には、DMA(A)14、DMA(B)15、命令キャッシュ16が接続される。DMA(A)14およびDMA(B)15は内部メモリ、即ちローカルメモリ(A)17およびローカルメモリ(B)18にそれぞれ接続され、これらメモリをダイレクトメモリアクセスする。バス13には、更にメモリインタフェース22を介してメインメモリ23およびROM24が接続される。

【0021】ローカルメモリ(A)17およびローカルメモリ(B)18はプロセッサ19に接続される。このプロセッサ19は、I/Oインタフェース21を介してマイコン25に接続される。このマイコン25には、センサ入力およびアクチュエータ出力が設けられている。

【0022】上記の構成において、ビデオカメラ11からビデオ信号を取り込む画像取り込み部およびこの画像取り込み部により取り込んだ画像に対して歪み補正などの処理を施す画像処理部が1チップ構成とされている。例えば、図1に示す破線で囲まれた回路部分、即ち、画像取り込み部としてのキャプチャユニット12および画像処理部を構成するDMA14、15、命令キャッシュメモリ16、ローカルメモリ17、18、プロセッサ19、I/Oインタフェース21およびメモリインタフェース22は1チップで構成される。

【0023】上述した画像歪み補正処理装置によると、ビデオカメラ11で撮影された画像はキャプチャユニット12によってバス13とメモリインタフェース22を経由しメインメモリ23上に配列として書き込まれる。プロセッサ19はメインメモリ23上にある命令をバス13を経由して命令キャッシュ16に取り込み、命令の実行を行う。

【0024】プロセッサ19はメインメモリ23上のデータを処理する場合、一旦ローカルメモリ(A)17またはローカルメモリ(B)18にDMA(A)14またはDMA(B)15を用いて転送し、ローカルメモリ

(A)17またはローカルメモリ(B)18をアクセスしてデータを処理する。

【0025】データをメインメモリ23に書き出す場合もローカルメモリ(A)17またはローカルメモリ

(B)18に書き出してからDMA(A)14またはDMA(B)15を起動してローカルメモリ(A)17またはローカルメモリ(B)18の内容をメインメモリ23に転送する事によってメインメモリ23に書き込みを行う。

【0026】ROM24はシステムのプログラムおよびデータを格納しておくために使用される。処理の結果はI/Oインタフェース21を使用してマイコン25とインタフェースを行う。マイコン25は、各種センサーからの入力およびプロセッサ19からの出力を用いて結果をアクチュエータに出力する。画像監視をするようなシステムではアクチュエータへの出力は警報音、警報ランプなどが考えられる。

【0027】一般にメインメモリ23の読み書き動作はローカルメモリ17、18より数倍から数十倍遅い。また、メインメモリ23は連続したアドレスをバーストとして読み出した方がランダムにアクセスするより多くのバンド幅が得られる。このメインメモリの特性を生かすためにメインメモリ23から一旦ローカルメモリ17、18にDMA14、15を利用して転送し、データにアクセスするのが高速化の手法として有効である。

【0028】ローカルメモリ(A)17およびローカルメモリ(B)18はいわゆるダブルバッファとして使用する事が可能である。例えば、DMA(A)14によりメインメモリ23からローカルメモリ(A)17に転送中にプロセッサ19はローカルメモリ(B)18の内容を処理し、つぎにDMA(B)15によりメインメモリ23からローカルメモリ(B)18に転送中にプロセッサ19はローカルメモリ(A)17の内容を処理する。このように、DMAと処理をオーバーラップさせる事により高い処理効率が得られる。

【0029】歪み補正は次のアルゴリズムに従って行われるが、歪みを伴った画像から1ドットずつ画像が読み込まれ、そのドットを歪み補正を済ませた配列に書き込んでいく。書き込み時には周辺のドットとの重みつき平均値をとることにより量子化による誤差の影響を押さえている。

【0030】

7

```

for(i=0; i<400; i++)
{
  for(j=0; j<640; j++)
  {
    xx=dis_x[i][j]
    yy=dis_y[i][j]
    img[i][j]=(dist1[i][j]*input[yy][xx]
               +dist2[i][j]*input[yy][xx+1]
               +dist3[i][j]*input[yy+1][xx]
               +dist4[i][j]*input[yy+1][xx+1])
  }
}

```

上記のアルゴリズムにおいて、歪み付き画像がinputであり、歪み補正後の画像がimgである。さらにdist1, dist2, dist3, dist4は重みで、dist_x, dist_yは歪みを補正するための配列である。さて、歪みを補正する上記アルゴリズムにおいて、dist_x, dist_yはカメラおよびレンズに依存し、画像には依存しない事を利用すれば、予め画像に依存しないパラメータをローカルメモリ(A) 17またはローカルメモリ(B) 18にDMA転送しておいて使用する事が可能である。ところで、ローカルメモリの容量には限りがあるので配列をすべて転送しておくわけにはいかない。ローカルメモリの容量に応じて例えば1行分の配列データだけを一度に処理するように制限する事が必要である。もちろん、ローカルメモリの容量が大きいときは例えば10行分のデータを一度に処理しても良いし、ローカルメモリの容量が小さいときは2分の1行のデータを処理するなどしても良い。以下では説明を簡単にするために1行分のデータを一度に処理する場合について説明する。

【0031】jを行方向に0から639まで変化させた場合のdist_xおよびdist_yの値により元画像inputからデータが読み出されるが、1ドットずつ読み出したのではメインメモリの効率が悪いので、jが0から639まで変化する間に読み出されるinputのドットを含むようなDMAの設定を予め求めておく。例えば、(xx, yy)が図2のように変化した場合、DMAを行う範囲としては図2のxが0から639, yがy1からy2だと十分である。しかしながら、ローカルメモリ容量並びに転送容量を小さくするためにはさらにDMAを行う範囲を陰線のように設定すると節約ができる。

【0032】このように陰線部だけを転送して節約を行った場合、DMA転送を行うには陰線のブロックの数だけ、複数のコマンドをDMAに与えなくてはならない。一般にDMAはメモリのバースト長などを配慮して行われるので、DMAの最小のサイズは決まっている事が多い。図2の場合には(xx, yy)の軌跡がDMAの最小サイズに満たない時はDMAの最小サイズを適用して転送する事とする。また、図2の場合にはDMAは11回必要になっている。この複数のDMA転送をプロセッサ19がすべて設定するのではプロセッサ19に対する負荷が重い。そこで、本発明ではディスクリプタを用いてD

8

MAを行うことにしている。ディスクリプタはローカルメモリに格納しておくのが一般的である。ディスクリプタは例えば図3(a)の形式を持っている。この形式でメモリアドレスは転送元のメインメモリ23からの転送を開始するためのメインメモリのアドレス、ローカルメモリアドレスは転送先であるローカルメモリの開始アドレス、長さは転送のサイズでバイト単位である。このディスクリプタがローカルメモリ17, 18上に連続して複数置かれていて、ディスクリプタの最後の終了フラグが1に設定されているとそのディスクリプタでDMAは終了する。

【0033】図3(b)にはこのディスクリプタを用いた例を示している。例では3つの転送を表しており、まず、メインメモリ23の1000番地から20バイトをローカルメモリ17または18の0番地から書き込み、次にメインメモリ23の1240番地から20バイトをローカルメモリ17または18の20番地から書き込む。最後に、メインメモリ23の1440番地から60バイトをローカルメモリの40番地から書き込んで終了フラグが1なので終了する。即ち、図2に示されるような画像のアドレスの配分がローカルメモリではリニアになるようにアドレス変換される。

【0034】さて、このようにディスクリプタをローカルメモリ17, 18上に置き、例えば一行分の処理を一回のディスクリプタでやれば、プロセッサ19は一回のDMAの起動で一行分のinput[xx][yy]に相当するデータをローカルメモリに持ってくる事が可能となる。しかしながら、このようなディスクリプタをその都度作成するのはプロセッサの能力を必要とするので、予めディスクリプタを作成して、メインメモリ23に格納しておき、これを順にローカルメモリ17, 18に転送して使用する事にしている。例えば、元の式でi行目に相当するディスクリプタはdesc[i]という配列でメインメモリ上に格納しておく。

【0035】最後に、このようにしてローカルメモリに転送させたデータを用いて重みをかけて平均をとる演算を行うが、DMA14, 15を用いてローカルメモリ17, 18に転送されたデータは演算に必要なデータより多く転送されている。また、input[yy][xx]に相当するデータがローカルメモリ17, 18上のどのアドレスにDMA転送されたかを求めなくてはならない。この演算は予め行っておく事が可能であり、演算した結果をLM_address[i][j]という配列としてメインメモリ23に格納しておき、必要な時にDMA転送でローカルメモリ17, 18に転送し、この値を使って演算を行う。LM_addressは4つの要素を持つ構造体で、LM_address.1がinput[yy][xx]のローカルメモリ17, 18上のアドレスを示し、LM_address.2がinput[yy][xx+1]のローカルメモリ17, 18上のアドレスを、LM_address.3がinput[yy+1][xx]のローカルメモリ17, 18上のアド

レスを、LM_address.4がinput[yy+1][xx+1]のローカルメモリ上のアドレスを示している。なお、input[yy][xx]、input[yy][xx+1]、input[yy+1][xx]、input[yy+1][xx+1]はローカルメモリにおける4つのドットをそれぞれ表し、これらドットは上、下、右、左にそれぞれ配列されている。

【0036】上記のDMA動作および処理は図4に示されている。即ち、図4に示すように、まずプロセッサ19はdist1、dist2、dist3、dist4をローカルメモリ

(A) 17に転送するようDMAを設定し、DMAを開始すると同時にローカルメモリ(B) 18の演算処理を始める。DMAが終了すると、終了割り込みがかかるので、次にプロセッサはdesc[i]をローカルメモリ(A) 17に転送するようDMAを設定し、ローカルメモリ(B) 18の演算処理は継続する。

【0037】このDMAも終了すると、終了割り込みがかかるので、今度はinput[yy][xx]、input[yy+1][xx]、input[yy+1][xx+1]、input[yy][xx+1]をdescに従ってローカルメモリ(A)に転送するようにDMAを設定し、ローカルメモリ(B)の演算処理は継続する。このDMAが終了すると割り込みがかかり、LM_addressをローカルメモリ(A)に転送するようセットしてローカルメモリ(B)の処理を継続する。ローカルメモリ(B)の処理が終わったらLM_addressのDMA転送が終了しているかどうか確認し、最後に、結果をローカルメモリ

* (B) からメインメモリに書き出すようにDMAを行う。

【0038】以上で、ローカルメモリ(A)にDMA転送している間にローカルメモリ(B)に転送された内容进行处理するダブルバッファ方式の処理の流れの説明を終える。ダブルバッファで処理するので、次は、ローカルメモリBにDMA転送している間にローカルメモリ(A)を処理するようにしている。

【0039】以上で説明した例は画面上の横一行を一度に処理しようとしているが、ローカルメモリの容量の関係で、1行を2度以上に分けて処理する事も可能だし、ローカルメモリの容量が大きければ1度に2行以上をDMAして処理する事も可能である。

【0040】以上、演算に必要なデータをローカルメモリ上にコピーし、ローカルメモリをダブルバッファとして使用する方法について述べてきた。なお、図1のシステム構成ではローカルメモリが物理的に二つ存在するように示されているが、一つのローカルメモリを分割してダブルバッファとして使用しても本発明の趣旨を逸脱しない。

【0041】次に、予め作っておくdescおよびLM_address.1、LM_address.2、LM_address.3、LM_address.4の作成方法について述べる。これらの配列は、以下のようなアルゴリズムによって作成される。

【0042】

```
for(i=0;i<400;i++){
  LM_START=ローカルメモリのdescを入れておく先頭番地にセット
  for(j=0;j<640;j++){
    xx=dis_x[i][j]
    yy=dis_y[i][j]
    if (xx,yyがdesc[i]の中に含まれなかったら) {
      desc[i]にxx,yyを含むようなブロックを付け加え、
      LM_STARTをブロック長だけインクリメントする}
    if (xx+1,yyがdesc[i]の中に含まれなかったら) {
      desc[i]にxx+1,yyを含むようなブロックを付け加え、
      LM_STARTをブロック長だけインクリメントする}
    if (xx,yy+1がdesc[i]の中に含まれなかったら) {
      desc[i]にxx,yy+1を含むようなブロックを付け加え、
      LM_STARTをブロック長だけインクリメントする}
    if (xx+1,yy+1がdesc[i]の中に含まれなかったら) {
      desc[i]にxx+1,yy+1を含むようなブロックを付け加え、
      LM_STARTをブロック長だけインクリメントする}
    desc[i]からxx,yyのローカルメモリのアドレスを求め
    LM_address.1[i][j]とする。
```

【0043】

```
desc[i]からxx+1,yyのローカルメモリのアドレスを求め
LM_address.2[i][j]とする。
```

【0044】

```
desc[i]からxx,yy+1のローカルメモリのアドレスを求め
LM_address.3[i][j]とする。
```


【0045】

desc[i]からxx+1,yy+1のローカルメモリのアドレスを求め
LM_address.4[i][j]とする。

【0046】

}
)

上記のアルゴリズムではブロックとしてDMAの最低のサイズをとっている。もちろん、DMA転送のサイズが大きいほど転送の効率は高いので、上記アルゴリズムを短いサイズのDMA転送をマージする事も可能であるが、説明のため、もっとも簡単なアルゴリズムを記述した。

【0047】次に、他の実施形態の画像歪み装置を説明する。

【0048】この発明では、1画面分の画像データを効率よく処理するために画像データの転送と同時に処理を行うダブルバッファ方式が採用されている。これは、図1に示されるように2つのローカルメモリ17、18が設けられ、一方のメモリ17に画像が取り込まれているときには、他方のメモリ18の内容が処理され、これとは逆にメモリ18に画像が取り込まれているときには、メモリ17の内容が処理される方式である。しかしながら、このような方式は、処理の時間が1画面を取り込む時間より短い場合には問題がないが、処理時間が画像取り込みより長い場合には処理が終わった瞬間に画像が取り込み中であつた場合、取り込み中の画像がメモリに転送されるまでの時間、処理が待たされることになる。

【0049】そこで、この実施形態では、図5に示すように1つのメモリに複数の画像バッファを設け、リング状に管理することで常に最新の画像を処理できるようにしている。即ち、図5によると、バッファ始端およびバッファ終端はそれぞれバッファ領域の先頭と末尾を示している。最新のポインタは、現在、最新の取り込み済み画像の先頭を示している。キャプチャポインタは現在画像が取り込まれているバッファ領域の先頭を示している。つまり、キャプチャポインタで示されるバッファ領域は取り込み中であるので処理には適さず、最新ポインタで示されている領域が処理できることになる。

【0050】次に、この実施形態の画像歪み補正処理装置による処理を図6および図7を参照して説明する。

【0051】キャプチャユニット12は1画面分の画像をメインメモリ23に転送し終わると、この転送終了を終了割り込みとしてプロセッサ19に知らされる(S11)。この後、プロセッサ19は、最新ポインタを1画面分プラスする。このとき、ポインタがバッファ終端にあれば、それはバッファ始端に更新される(S12)。キャプチャポインタが新たな最新ポインタに更新されると、新たな画像取り込みが開始される。もしキャプチャポインタがバッファ終端であれば、バッファ始端に更新される(S13)。

【0052】このように複数のバッファを使用した割り込みルーチンは常に最新ポインタが最新の取り込んだ画像であることを保証している。

10 【0053】図7は、上記の割り込みルーチンを使用した画像処理の全体の流れを示している。これによると、装置がスタートされると、最新のポインタの領域から1画面分の画像がメインメモリからロードされる(S21)。このときロードされた1画面分の画像は図8

(a)に示されるような歪みを有する画像である。この歪み画像に対してプロセッサが歪み補正アルゴリズムに従って歪み補正を行う。この場合、プロセッサは歪みを伴った画像から1ドットづつ画像を読み込み、歪み補正を行い、補正後の画像は元のローカルメモリに書き込まれ、その後、メインメモリに転送される。この時に得られる補正画像は図8(b)に示されるようにリニアな画像となる。

【0054】

【発明の効果】本発明によれば、マイクロプロセッサを用いて高速に画像の歪みを補正する事が可能となる。これは、画像のローカルメモリへの転送量を1/10程度に押さえられる事で、少ないローカルメモリでも処理が行えることにより、高速なローカルメモリが使用できる事の効果によるものである。

30 【図面の簡単な説明】

【図1】本発明の一実施形態の画像歪み補正処理装置のブロック図

【図2】DMAで読み出す範囲を示す図

【図3】ディスクリプタを説明する図

【図4】DMAの動作を説明する図

【図5】内部メモリのフォーマットを示す図

【図6】プロセッサの割り込み処理を示すフローチャート図

【図7】画像処理の流れを示すフローチャート図

40 【図8】歪み画像および歪み補正画像を示す図

【符号の説明】

11…撮像カメラ

12…キャプチャユニット

13…バス

14、15…DMA

16…命令キャッシュメモリ

17、18…ローカルメモリ

19…プロセッサ

21…I/O

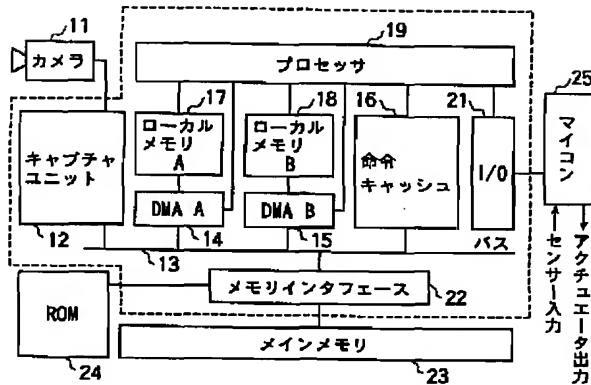
50 22…メモリインタフェース

23…メインメモリ
24…ROM

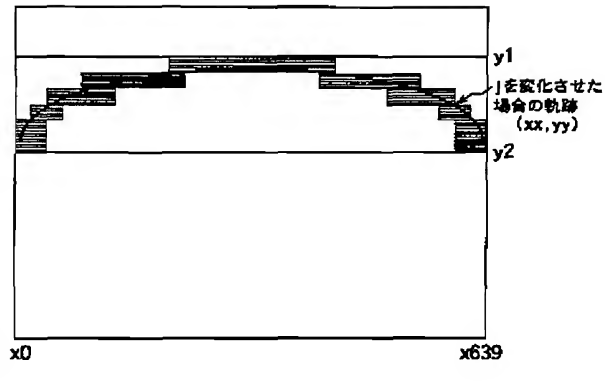
* 25…マイコン

*

【図1】



【図2】



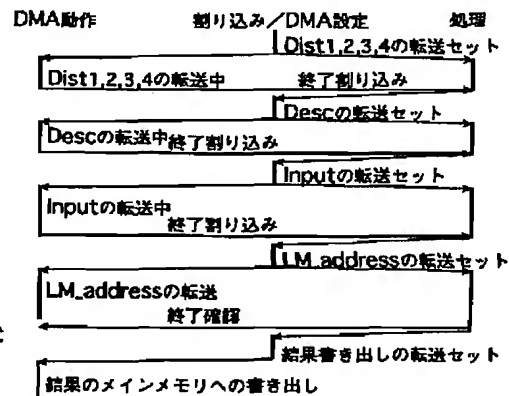
【図3】

31	0	1000
メモリアドレス		0
ローカルメモリアドレス		20
長さ		0
終了フラグ		1240
		20
		20
		0
		1440
		40
		60
		1

(a) ディスクリプタの形式

(b) ディスクリプタの形式

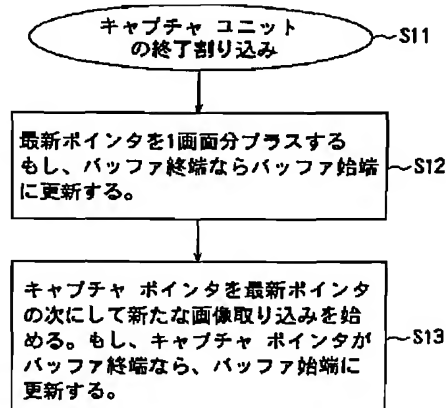
【図4】



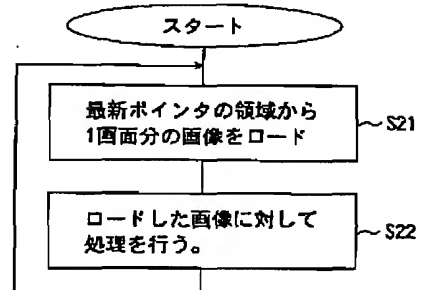
【図5】

バッファ A	
バッファ B	最新ポインタ
バッファ C	キャプチャ ポインタ
バッファ D	バッファ 終端

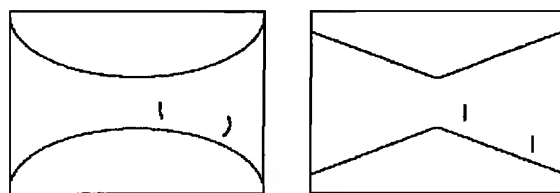
【図6】



【図7】



【図 8】



(a)

(b)